

DBMS

Date / /
Page _____

- What is DBMS & what is its utility? Explain RDBMS with ex.?

A DBMS stands for Database Management System, is a set of applications & programs that enable user to Create & Maintain Database. DBMS provide a tool or an interface for performing various operations such as Inserting, deleting, updating, etc. into a database.

DBMS helps to overcome problems like data inconsistency, data redundancy. Data is more organized in DBMS.

Eg (given DBMS) \Rightarrow file Systems, XML

Windows Registry, etc.

about tool 2: \Rightarrow MySQL, Oracle

(written) tools: MySQL, PostgreSQL, etc.

RDBMS = Relational Database Management System

DBMS stores data as files

RDBMS stores files stored Tables in the

form of Rows & Columns

- What is Database?

A Database is an organized, consistent and logical collection of data that can easily be updated, accessed and managed. It which consists of Records and Fields.

A Row is a simple entry in a table

Attribute or Column represents basic unit of data storage.

DBMS extracts data from a database in the form of queries.

3. Mention the issues with traditional file-based systems that makes DBMS a better choice?

→ The absence of indexing in a traditional file-based system.

Hence makes access control tedious & super slow.
Predicndancy, inconsistency as files have many duplicate and redundant data.

→ Changing one of them makes all of them inconsistent.

→ Data is unorganized in traditional file-system.

→ Lack of Concurrency Control.

→ Multiplexed operations can't be done in DBMS.

Integrity checks, data isolation, atomicity, security, etc are some other issues with traditional file system.

4. Few advantages of DBMS.

- Data sharing

- Integrity constraints

- Controlling redundancy in a database

- Data independence

- provides backup & recovery facility

- Data Security

5. Different languages in DBMS?

DDL: (Data definition language) (defines the data base)
e.g., (CREATE, ALTER, DROP, TRUNCATE, RENAME)

DML: Data manipulation language

e.g., SELECT, UPDATE, INSERT, DELETE, etc

based on DCL (Data Control Language) :-

Deals with user permissions and controls managing in the database system.

Eg :- GRANT, REVOKE

TCCL (transactional control language) :-

deals with transaction of the database.

Eg :- COMMIT, ROLLBACK, SAVEPOINT

What is meant by ACID properties in DBMS?

ACID ensures a safe & secure way of sharing data among multiple users.

A - Atomic.

Change in Both

A + B = C
Debit Credit

Ensures total value remains same

Atomicity :- All changes to data must be performed successfully.

and not at all.

Consistency :- Data must be in a consistent state before and after the transaction.

and after.

Isolation :- No other process can change the data while the transaction is running.

Durability :- The changes made by a transaction must persist.

Atomicity :- This reflects either executing the whole query or executing nothing at all, which implies an update occurs in a database then that update should either be reflected in the whole database or not reflected at all.

Consistency : Data remains consistent before and after a transaction in a database.

Isolation : Transaction is occurring independently of the others.

It implies the state of an ongoing transaction doesn't affect the state of another ongoing transaction.

Durability : This ensures data is not lost in cases of a system failure or restart and is present in the same state as it was before the system failure or restart.

7. Are NULL values in database the same as that of blank space and zero?

No, they are different. NULL represents unknown.

NULL Represents Unknown
0 Represents student has taken 0 courses

and which set of map the address

8. What is Data Warehousing? (DBMS interview question)

The process of collecting, extracting, transforming and loading data from multiple sources and storing them is known as data warehousing.

can be considered as central repository

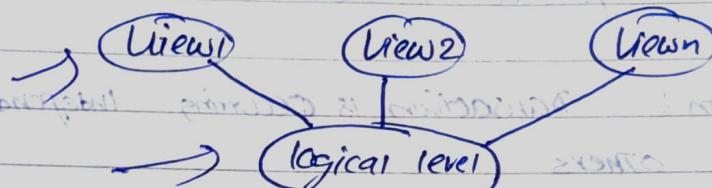
Data ware house helps in Analysis, Reporting, Data Mining.

9. Different levels of data abstraction in a DBMS.

process of hiding irrelevant details from users is known as data abstraction.

Data abstraction can be divided into 3 levels

entity is abstracted into View



Three levels of data abstraction

at top level of abstraction just a collection

at bottom level lowest level, p managed by DBMS.

contains how data storage + description

hidden from developers, user, admins

Conceptual level logical level

what data is stored behind the database and what is the relationship between the data points

External or view level

describes only part of the database and hides details of the table schema and its physical storage from the user

e.g. result of a query is an example of external level of data abstraction

What is meant by entity relationship model

explain the terms: entity, entity type, entity set

DBMS

It's a diagrammatic approach to a database

where real-world objects are represented as entities and relationships

a record

Entity is defined as The real-world Object having attributes that represent characteristics of That particular object.

Eg: Student, an employee

entity type is collection of entities that have the same attributes.

entity set is Set of all the entities present

(in a specific entity type) in a database.

→ Set of entities

Different types of relationships amongst tables in a DBMS

1-1, 1-many, Many-Many, Self-referencing

⇒ One to one:

when a particular row in table x is linked to a singular row in table y



⇒ one to many Relationship:

when a particular row in table x is related to many rows in table y



⇒ Many-Many Relationship

when multiple rows in table x can be

linked to multiple rows in table y



⇒ Self Referencing Relationship

when a particular row in the table x is associated with the same table.



Q12. Explain the difference between intension & extension of a database.

Ans. Intension \rightarrow popularity known as database schema

\rightarrow Defines the description of the database

\rightarrow Specified during the design of DB

Extension \rightarrow Measure of the no. of types present in the database at any given point

Eg: Snapshot of DB

Changes according to the time.

Q13. Explain the difference b/w DELETE & TRUNCATE Commands.

DELETE

\rightarrow To delete rows from a table \rightarrow It removes complete table based on conditions

\rightarrow Deletes only rows \rightarrow removes complete data

\rightarrow Can be rolled back \rightarrow can't be rolled back

\rightarrow It maintains a log before deleting Hence \rightarrow no log is maintained

\rightarrow It's slow \rightarrow fast

Q14. What is lock?

Lock is a mechanism to protect a shared piece of data from getting updated by two or more databases.

When a row is accessed it is locked in other DB student can't access it.

(Shared) \rightarrow (Exclusive) \rightarrow (Shared)

Shared-lock : (S)

- ⇒ it is for reading a data
- Many transactions may hold a lock (shared lock)
- Multiple transactions are allowed to read the data that are in a shared lock.

Exclusive lock : (X)

- acquired before performing write operations
- it won't allow more than 1 transaction
- prevent (inconsistency) in the DB.

15. Normalization & Process of Reducing Redundancy

- by organization data into multiple database.
- leads to better usage of disk spaces
- Helps to maintain Integrity

Denormalization = Reverse process of Normalization.

Combines the tables which have been normalized into a single table.

Join operation is used in denormalization.

⇒ data retrieval is faster in denormalization.

16. Different types of normalization forms in DBMS

INF = Simplest form of normalization.

⇒ Simpliest Normalization process

⇒ every column must have Single Value

⇒ value must be atomic.

⇒ Duplicate Column should be removed

⇒ each row must be identified with a

Unique Column.

2-NF \dagger (2-table)

Satisfy all the conditions in 1-NF.

Condition: Non-prime attribute of the table should be fully functionally dependent on the primary key.

~~3NF~~ \ddagger 3rd Normal form.

Requirement: Table should satisfy 2-NF rule.

Condition: There is no transitive functional dependency.

Condition: Non-prime attribute is not functionally dependent on any attribute in the same table.

Transitive dependency means) or Frame \rightarrow Fee Structure
 depending on primary key via

Supernormal anomalies

Insertion anomaly

Deletion anomaly

Update anomaly

BCNF \dagger Boyce-Codd Normal Form

Advanced version of 3NF

3-SNF

For every functional dependency A on B

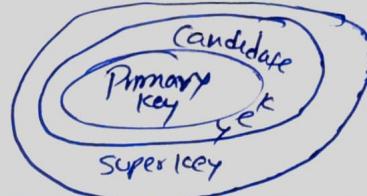
$(A \rightarrow B)$ A should be Super key.

Example: Primary identity

As this: S-ID, Name, age, emp-ID, Dept

S-ID	NAME	age	emp-ID
1			

Emp-ID	Dept



Data _____
Page _____

17. Explain different types of keys in a DB?

Primary key \Rightarrow Uniquely identifies a record
foreign key \Rightarrow helps to take values from another table. Helps to JOIN.

unique key = Primary key but allows NULL values

alternate key = except primary key

Candidate key = Super key

group of multiple key uniquely identifies values in a table. (Identifies attributes)

Composite key = key refers to a combination of two or more columns

Making two Primary key into 1 as unique

4-NF: follows BCNF has no multi-valued

dependency

multivalued dependency

ID	Course	Hobby
21	Computer	Dancing
21	Math	Singing
21	Chemistry	Dancing
34		

ID	Course
21	Comp
21	Math
34	Chem

ID	Hobby
21	Dancing
21	Singing
34	Dancing

5-NF: \Rightarrow Must follow 4NF \Rightarrow ~~4NF~~

should not contain any join dependency
Joining should be lossless.

\Rightarrow Project-Join

3-Tier

Sub	Lec	Sem
Comp	Anushka	1
Math	John	1
Math	Akash	2
Chemistry	Praveen	1

(1)	Sub	Lec	(2)
Sem	Comp	Anushka	Sub
1	Math	John	Lec
2	Chem	Akash	Comp
	MATH	Praveen	Math

Sem	Lec
1	Anushka
1	John
2	Praveen
2	Akash
	John

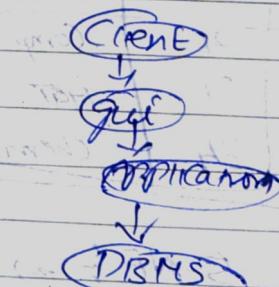
18) 2-tier

- Client-Server architecture
- Client directly communicate with database

Eg: Contact Management System or Railway System etc.

3-tier architecture

More Secure & accessible



websites on the internet

Ex: Google, YouTube, Facebook, etc.

client -> application -> DBMS

SQL Interview Questions:

Data _____
Page _____

QUESTION

1)

What is Database?

A database is an organized collection of data, stored, retrieved, manipulated, controlled.

2)

DBMS is a software responsible for the creation, retrieval, updation, & management of database.

3)

What is SQL?

Structure Query language. Standard language for RDBMS.

4)

SQL is used for MySQL

for retrieving & manipulating RDBMS system like database.

SQL Server, Oracle, IBM

5)

Tables: Organized collection of data stored in the form of rows & columns attributes.

fields: Columns in a table are called as fields.

Rows = record

(a) Records

6)

Constraints in SQL?

Specifies the rules concerning data in the table using alter command.

- NOT NULL \Rightarrow restricts NOT TO INSERT NULL VALUES
- CHECK \Rightarrow Verifies 'Satisfy / not a Condition'
- DEFAULT \Rightarrow Assign Default Value
- UNIQUE \Rightarrow ensures Unique values to be inserted
- INDEX \Rightarrow Provides faster retrieval of records
- PRIMARY KEY \Rightarrow uniquely identifies each record.
- FOREIGN KEY \Rightarrow Referential Integrity

widely used,

- 7) INNER JOIN: Retrieves matching value
Left (Outer) Join: Retrieves left & matched records
Right (Outer) Join: Retrieves Right & matched left
Full (Outer) Join: No matched records is retrieved
Self Join: uses INNER OR Left JOIN
Recursive Join: same table is joined up to itself.
Cross Join: defines as Cartesian Product of the two tables (works like INNER)

- 8) INDEX: Quick look-up of data in a column
Unique INDEX: maintains data integrity
Non-unique INDEX improves query performance

- 9) Clustered & Non-clustered Index:
 Order of the rows corresponds to the order of data in the index.

- 10) clustered: linear access path

- 11) Data Integrity is assurance of accuracy & consistency of data

- 12) Query : request for data & information from database table.

Sub Query: Nested Query (dependent on main query)

Query or subquery: query

Correlated Sub Query

Non-correlated Sub Query

Not an Independent Query

Independent Query

Refers to the column in the table

Output is substituted in the main query

- 13) Common clause : where, Order By, GROUP BY, HAVING
14) UNION = Combines, MINUS = removes duplicates

- 15) What is cursor?

cursor is a control structure, used for traversal of records in a database.

e.g. DECLARE |
OPEN |
| FETCH |
CLOSE |
DEALLOCATE

DECLARE USING Variables

- 16) ALIAS (AS)

- 17) View? Virtual table of resultant SQL statement

- 18) Drop Remove TABLE, TRUNCATE : Structure won't get deleted

- 19) Aggregate Function.

returns scalar value

used with group by, HAVING

SELECT

e.g. Avg =

Min =

Max =

Sum = Calculated sum of

Count = Total no. of records

First last

Scalar Functions

Len()

UCASE()

LCASE()

Mid()

Concat()

Format()

ignores null value except count

- 20) User-Defined Functions

Scalar Function

returning single scalar value

returning

Table-valued Function

INLINE : 1 TABLE

Multi-statement : Multiple SELECT

Backened
→ row based

Q1) OLTP → Online Transaction Processing
It is a software for transaction oriented programs.

OLAP → Analytical Processing

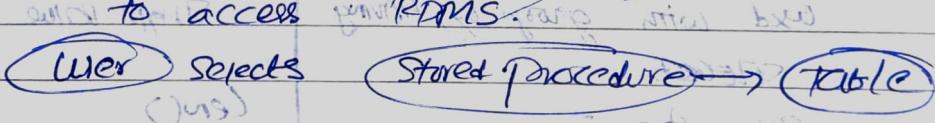
→ column based

uses aggregations

Q2) Categorization: Set of rules that determined how a data is stored.

- ✓ Case Sensitivity → A & a are differently treated
- ✓ Accent Sensitivity → á, à, á, á
- ✓ Kanji Sensitivity → おとこ
- ✓ Bottom Sensitivity → Hiragana, Katakana

Q3) Stored Procedure & Subroutine are available



Q4) An empty table like another file structure

~~Select * INTO Student's copy From Students
WHERE i=2;~~

~~Two copies created with command~~

~~Student's copy~~

~~student's backup file~~

~~student's copy~~

~~most secure storage~~

~~student's~~

~~secure~~